ANDREW P. BRIDGES (CSB No. 122761)
abridges@fenwick.com
ILANA RUBEL (CSB No. 221517)
irubel@fenwick.com
KATHLEEN LU (CSB No. 267032)
klu@fenwick.com
FENWICK & WEST LLP
555 California Street, 12th Floor
San Francisco, CA 94104
Telephone: 415.875.2300
Facsimile: 415.281.1350

Attorneys for Defendants
GIGANEWS, INC. and
LIVEWIRE SERVICES, INC.

UNITED STATES DISTRICT COURT

CENTRAL DISTRICT OF CALIFORNIA

WESTERN DIVISION

PERFECT 10, INC., a California Corporation ,

Plaintiff,

v.

GIGANEWS, INC., a Texas Corporation; LIVEWIRE SERVICES, INC., a Nevada Corporation; and Does 1 through 100, inclusive,

Defendant.

GIGANEWS, INC., a Texas Corporation; LIVEWIRE SERVICES, INC., a Nevada Corporation,

Counterclaimants,

v.

PERFECT 10, INC., a California Corporation,

Counterdefendant.

Case No. 11-cv-07098-ABC (SHx)

DECLARATION OF JOHN LEVINE, PH.D., IN SUPPORT OF DEFENDANTS' OPPOSITION TO MOTION FOR PARTIAL SUMMARY JUDGMENT

I. QUALIFICATIONS AND ASSIGNMENT

1. I have over 30 years experience in computing, computer and communications networks, the Internet, and Usenet. My current curriculum vitae is Exhibit A to this declaration.

2. I have been an active participant in Usenet continuously since 1981. Since 1986 I have been the moderator of the *comp.compilers* Usenet newsgroup, which discusses technical topics related to computer compiler software. I have run my own Usenet news server since the 1980s. I have also developed and continue to run group moderation software for a variety of other newsgroups including soc.religion.unitarian-univ (Unitarian Universalists), and misc.transport.air-industry (commercial aviation).

3. I have written or co-authored a variety of technical computer books. These include *The Internet for Dummies*, 1st through 13th editions in 1993-2011; *Linker and Loaders*, published by Morgan Kauffmann in 2000; *lex & yacc*, published by O'Reilly in 1992; *Programming for Graphics Files in C and C++*, published by Wiley in 1994; *Understanding Javelin Plus*, published by Sybex in 1987; and *qmail*, published by O'Reilly in 2004.

4. Since 1987 I have been a self-employed writer and consultant, formerly doing business under the name I.E.C.C. and now under the name Taughannock Networks (pronounced ta-GONN-ick).

5. From 1979 to 1981, I worked part-time and from 1981-84. I worked full-time for INTERACTIVE Systems, Santa Monica CA and Boston MA, a vendor of UNIX operating systems and related software. While working part-time I wrote a compiler that I believe to be the first commercially available version of the Fortran 77 programming language. From 1982 to 1984, I was the operating system architect for AIX 1.0, a version of UNIX that INTERACTIVE wrote under contract for IBM for their RT PC workstation, and wrote or supervised the writing of many programming tools for that system.

6. From 1984 to 1987, I was a Senior Member of the Technical Staff of Javelin Software, Cambridge MA, where we wrote Javelin and Javelin Plus, award winning desktop modeling programs. While there my duties included development of the Javelin and Javelin Plus products, maintenance of compiler-related programming tools and liaison with our compiler vendor.

7. From 1993 to 1996, I was the editor and publisher of *The Journal of C Language Translation*, a quarterly technical journal about technical and standardization issues in the C programming language.

8. I have spoken at many professional, trade, and government fora. I have served on advisory boards related to consumer Internet issues at companies ranging from Orbitz, one of the big three on-line travel agencies, to Habeas, a small anti-spam certification startup in Palo Alto CA.

9. I was a member of the At-Large Advisory Committee for the Internet Corporation for Assigned Names and Numbers (ICANN) from 2005 to 2007.

10. I was the chair of the Internet Research Task Force (IRTF) Anti-Spam Research Group. Since 1997, I have been a board member and since 2006 President of the Coalition Against Unsolicited Commercial Email (CAUCE), an Internet user advocacy group, and I currently run the Network Abuse Clearinghouse, a free service that helps Internet users report and deal with abusive online behavior.

11. I have served as an expert witness in a number of cases and have provided live testimony at trial in four cases in the past four years.

II. SUMMARY OF CONCLUSIONS

I have reviewed the Declaration of Norman Zada that Perfect 10 submitting in support of its motion for partial summary judgment ("Zada Declaration"). In it, he draws many incorrect conclusions, many of which stem from incorrect assumptions or assertions of fact. Below, I summarize my main conclusions:

- Message-IDs function as addresses for locating Usenet messages. The Zada

Declaration misunderstands and wrongly explains their role.

- A Usenet service provider must have a Message-ID in order to remove a Usenet message from its servers.
- A copyright holder who has found an infringing message may easily view the Message-ID of that message in order to furnish it in a notice to a Usenet service provider.
- Searches are not a reasonable or effective substitute for Message-IDs in locating particular messages or content on Usenet.
- Identifying a person who posts a message to Usenet requires particular information that is not immediately apparent in a message, and generally a Usenet provider can identify senders of messages only when they are the provider's own customers.

III. OVERVIEW OF USENET

12. Usenet is the Internet's distributed bulletin board system. The Usenet operates side by side with better known services such as e-mail and the World Wide Web but is different from them.

13. The Usenet functions differently than other Internet services, and understanding its basic structure is important to understanding the facts in dispute here as well as problems in the Zada Declaration.

14. The Usenet consists of a vast network of servers all over the world. The material on Usenet consists of *messages* organized in *newsgroups.* Usenet messages have a structure deliberately similar to email. A principal difference is that Usenet users address their messages to newsgroups rather than particular recipients.

15. Usenet messages consist of a header and body. The header consists of several fields. I will discuss in greater detail the most relevant header fields below.

16. Usenet messages come from users. A user generates a message and posts it to the Usenet by posting it to one Usenet server of his or her Usenet service provider.

17. Usenet is designed to that the first Usenet server to receive a message will automatically store and forward the message to other servers with which it has a connection, and then those servers will store and forward the message to the servers they have a connection with, and so on. In that fashion, a message from a single user will generally reach the entire Usenet and become available to other users.

18. After a user posts a message to Usenet, and that user's service provider forwards it to other Usenet services, another user may retrieve the message from a server of that other user's Usenet service provider. That server may reside many links down the chain from the originating server.

19. Usenet servers are designed to operate as much as possible in entirely automated fashion. As a Usenet operator I routinely go several months without having to do any manual management of my own server.

20. Most users use software called newsreaders to post and retrieve messages. Newsreaders are specialized browsers for the Usenet that reside on the user's computer, not the Usenet server. Some newsreaders are part of integrated message handling programs that handle e-mail, Usenet, and other messaging, such as Microsoft's Outlook Express and Mozilla Thunderbird. Other newsreaders such as Forte Agent are specialized for Usenet and have features to help users navigate the Usenet.

21. All newsreaders communicate with news servers using Network News Transfer Protocol (NNTP) and a subset of it called Network News Reading Protocol (NNRP). Servers do not post new messages on their own; they only store messages received from users, either directly from a service's own customers or indirectly from other service providers' customers.

22. Newsreader software programs typically provide great flexibility to users when they post messages. The user can control the the name and e-mail address on the From line as well as the Subject, Newsgroup(s), and contents of the message. To post a message, a user composes the message in the newsreader, connects to his or her news server if not already connected, and tells the newsreader to post it by transmitting it by NNRP to the server. The server checks the message for technical correctness, adds headers such as Date and Message-ID if the user's newsreader software has not already added them, places the message in its message store, and schedules the message to be transmitted to the server's peers.

IV. ONLY MESSAGE-IDS UNIQUELY IDENTIFY USENET MESSAGES.

23. Each message has a Message-ID which acts as a unique identifier for the message. Message-IDs function as addresses for locating Usenet messages. The Zada Declaration misunderstands their role.

24. Each message has a *header*, consisting of structured information about the message, and a *body*, with structured or unstructured text that constitutes the content of the message.

25. This is a typical Usenet message, one that I sent in July 2011:

Path: mx04.eternal-september.org!eternal-september.org!feeder. eternal-september.org!feeder.er je.net!newsfeed.straub-nv.de!news.glorb.com!transit3.readnews.com!news-out.readnews.com!news-xxxfer.readnews.com!news.misty.com!news.iecc.com!not-for-mail
From: John Levine <johnl@iecc.com>
Newsgroups: comp.arch
Subject: Re: Arcane 704 history question
Date: Tue, 5 Jul 2011 17:56:15 +0000 (UTC)
Organization: Taughannock Networks, Trumansburg NY
Lines: 17
Message-ID: <iuvj7v$20u7$1@gal.iecc.com>
References: <iut5p3$24l6$2@gal.iecc.com> <s1lae8-olf.ln1@ntp6.tmsw.no> <0360d0b5-b201-42e8-9dab-322ec4da46e2@u19g2000vbi.googlegroups.com>
NNTP-Posting-Host: news.iecc.com
X-Trace: gal.iecc.com 1309888575 66503 64.57.183.58 (5 Jul 2011 17:56:15 GMT)
X-Complaints-To: abuse@iecc.com
NNTP-Posting-Date: Tue, 5 Jul 2011 17:56:15 +0000 (UTC)
Cleverness: some

X-Newsreader: trn 4.0-test77 (Sep 1, 2010)
Originator: johnl@joyce.lan (John L)
Xref: feeder.eternal-september.org comp.arch:16975

70x Fortran stored arrays in reverse column major order. It'd work, for a highly constrained version of work.

Apparently the Mark III had a sort of similar thing that dynamically adjusted the last couple of digits of the address, so your arrays had to be 10 or 100 elements on a side.

R's,
John

26. As one can see, one of the elements of the message is a "Message-ID." The program that a user used to create the message or the first news server to which the message is submitted, assigns a unique Message-ID to distinguish the message from all other Usenet messages.

27. The form of the Message-ID is similar to that of an e-mail address, with two parts separated by an @ sign. The part after the @ sign is an Internet domain name, typically that of the computer or service to which the message was submitted, and the part before the @ sign is a pseudo-random string of characters designed to be unique. In the example above, the Message-ID is "iuvj7v$20u7$1@gal.iecc.com", because the computer on which my Usenet server software runs is called gal.iecc.com. News servers also use the Message-ID to recognize duplicate messages, as I explain below.

28. Each Usenet server keeps a message store or repository, including messages posted by its own users and those received from other servers. Each server has relationships with one or more other Usenet servers. To operate efficiently, and to forward messages throughout Usenet without causing redundant transmissions, Usenet requires several technological processes and features. Each server uses a "flooding" technique to pass all the messages it has to all of its peer servers. To avoid wasted traffic, each message has a Path: field in the header, which lists all of the servers through which the message has already passed, so a server

knows not to resend a message to servers already in the message's path. The servers also use Message-IDs to avoid wasted traffic, with the sending server first sending the Message-IDs of messages, and the recipient server telling the sender which of those messages it doesn't already have. This flooding technique is quite effective, and a message typically is distributed among Usenet servers within an hour or less of when it is posted. I have arranged for my own Usenet server to peer with Giganews, and I have verified that its system peers in the standard way.

V. A USENET PROVIDER NEEDS A MESSAGE-ID TO REMOVE A PARTICULAR MESSAGE FROM USENET.

29. If a copyright holder provides a Message-ID of a message to be deleted to a Usenet server operator, the operator uses a maintenance command to delete the message. On my server, for example, the command is:

```
ctlinnd cancel 'abc@xyz'
```

where abc@xyz is the Message-ID.

30. Public server operators generally have a published policy describing under what circumstances they will remove messages from their message store. Since every message has a unique Message-ID and the message store is invariably indexed by Message-ID, it is straightforward to remove a message or group of messages if their IDs are known. Lacking the Message-ID, it is difficult or impossible to find a message, since it would involve searching a message store potentially containing many millions or billions of messages.

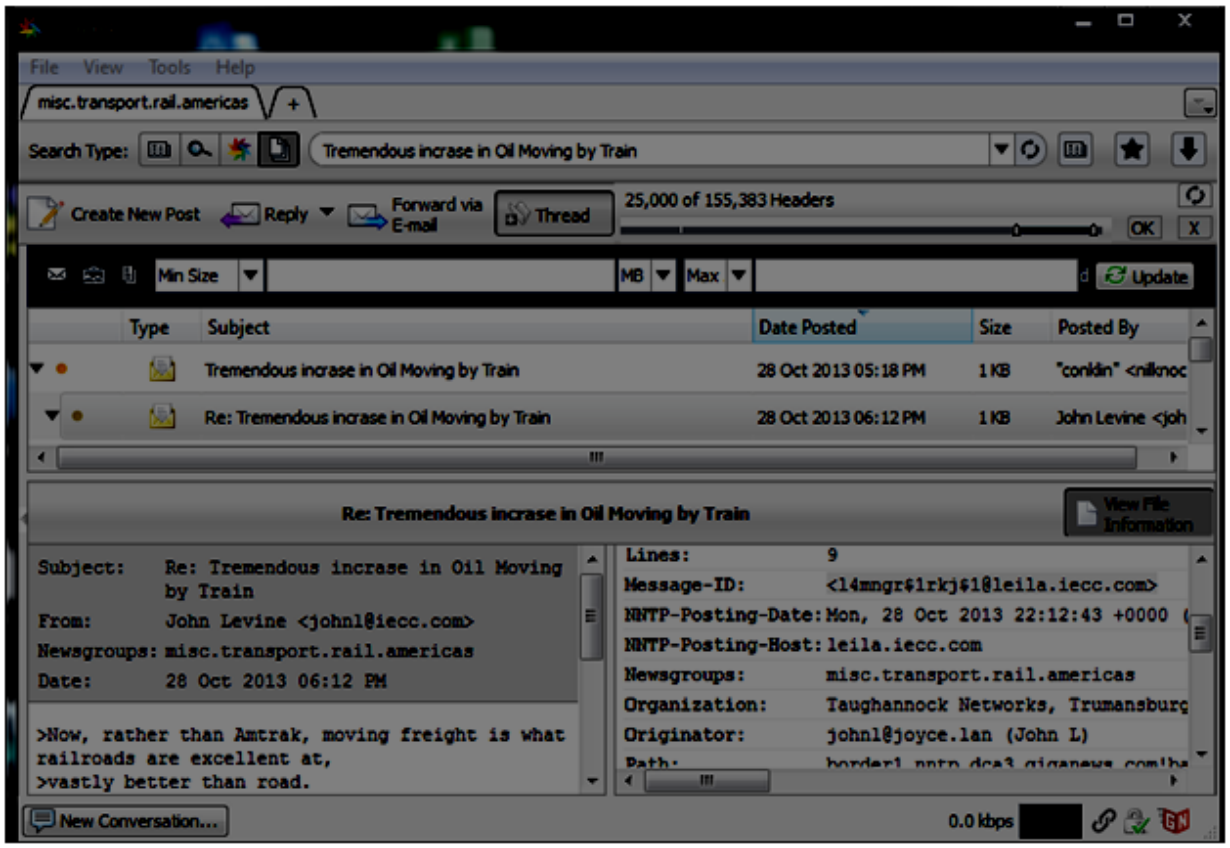
31. Without the Message-ID, it is difficult and frequently impractical for a server operator to locate a particular message.

VI. A COPYRIGHT HOLDER CAN READILY OBTAIN MESSAGE-IDS FOR INFRINGING MESSAGES THAT IT DISCOVERS.

32. Individuals use newsreader programs to read Usenet messages and post new messages. I used the Mimo newsreader program to look for some

messages that interested me. A copyright holder could do the same, trying different search terms to explore what messages are on Usenet.

33. After searching and getting a results list with some messages, I clicked the Mimo "View File Information" button which displays message headers including the Message-ID. Below is a screenshot of search results with a window displaying the message headers. If I wanted to collect the IDs of several messages, I could easily use the usual copy and paste process to copy the ID to a document. The copyright holder could go through the same steps to retrieve these Message-IDs, and would better positioned to do so as the copyright holder knows at the outset exactly what messages he or she wishes to identify for removal.



VII. SEARCHES CANNOT SUBSTITUTE FOR MESSAGE-IDS TO LOCATE PARTICULAR MESSAGES OR CONTENT.

34. Servers typically index a message's Message-ID, the date it was posted, the date it expires, and its newsgroup(s). Of those elements, only Message-ID can identify a particular message.

35. Some third parties, such as nzbindex.nl, create indexes of subject lines across newsgroups. But those indexes exist independently of the message storage at the Usenet server level. They may be helpful sources for identifying information that may be available on the Usenet, but they are unauthoritative and unreliable.

36. If a copyright holder provides only the subject of the message in a takedown request, servers have no built-in way to delete by subject, so the operator would have to find the Message-IDs first by running a search using a newsreader. This assumes that the search is successful and accurate; as noted above Usenet subject indexes are often incomplete, and they also change frequently, so it can be at best a guess that the message found is the one that the copyright holder intended.

37. To run a search, the Usenet operator could do a subject search in a Usenet newsreader. Newsreaders have features to scan through a newsgroup looking for particular subjects, authors, and other message characteristics. This would be the same steps the copyright holder could take to look for the same message(s). If a newsreader search can correctly identify the message or group of messages the copyright holder intended, the associated Message-IDs must be extracted and then the messages can be deleted.

38. If the copyright holder merely provides a copy or a description of an infringed work, I know of no practical way to identify the message or messages corresponding to the work. Usenet servers provide no practical way to look

through the bodies of messages, and even if they did, given the many possible ways a work might be encoded, there is no way to tell what to look for.[1]

39. Even if a copyright holder provides a file name of an image, again there is no practical way to look for it since Usenet servers do not index the file names that might appear within the encoding of a binary file within a message. By Usenet convention, filenames are often included in the subject lines of messages, but as described above, a subject search at best provides a guess about what message or messages the copyright holder intended. Many very different messages may have similar subject lines.

VIII. IT IS NOT DIFFICULT TO RETRIEVE A MESSAGE USING A MESSAGE-ID.

40. If a user had the Message-ID to a particular message and wanted to see if it was on a particular server, it is possible to retrieve it directly. First, there is a simple text-only free Internet browser "Lynx" available on any Linux or similar system that supports, among other things, using NNTP to retrieve a message by Message-ID. Furthermore, for a user with some programming skill, which Mr. Zada claimed to have in his declaration at ¶ 2, it is easy to write a small script to fetch a message's headers or body by Message-ID. (While preparing this document, I wrote one in about 15 minutes.) Once written, the program could be reused over and over again with more Message-IDs.

[1] Usenet has developed conventions for encoding pictures and other media as text, and Usenet servers treat every message as a text message. The encoding and decoding is a multi-stage process, performed by the user's newsreader program, not the server. Newsreader programs typically include functions to handle encoding and decoding and image display. If a Usenet message includes an image, the newsreader first uses a decoder to extract the image file then displays the image on the user's screen. I have never seen a Usenet news server that decoded image files, and I would be very surprised if any did. Usenet news servers are written to handle large volumes of messages as efficiently as possible, and there is no need to decode images to store or retrieve them, or to support connections to other servers and newsreaders. Hence it is unlikely that any Usenet news server could do any sort of message selection based on images in the messages. Usenet services have no more information about images in the messages than any ordinary user with a newsreader would have.

IX. IDENTIFYING A SENDER ON USENET, FOR PURPOSES OF APPLYING A REPEAT INFRINGER TERMINATION POLICY, REQUIRES INFORMATION NOT APPARENT IN USENET MESSAGES.

41. Usenet users have traditionally used a wide range of genuine and false return addresses in the From: lines of their messages, and news servers have never attempted to require that users use a "real" return address. In the early 1990s, as spam (junk e-mail) started to be a problem, Usenet users found that spammers were mechanically harvesting the return addresses in Usenet messages as a source of target addresses to spam. In response, many Usenet users started disguising ("munging") their return addresses. A munged address might include text that a human would know to remove, such as joesmith@hotmailREMOVETHIS.com. Or the address on the From: line might be entirely false, such as notme@nomail.invalid, with the user's actual address in munged form somewhere in the message body.

42. At this point, it is debatable whether spammers bother to look for addresses in Usenet messages, since they now have so many other sources, but the tradition of munged and false return addresses continues, due to fear of spammers, online stalkers, or just not wanting to be bothered by e-mailed responses. Although I personally have always used real return addresses in my Usenet messages, I know many people who routinely use munged or fake addresses. Some hide their real identity, using only cryptic pseudonyms, others make no attempt to hide their real identity, but just don't want to publish an e-mail address.

43. Every server has some method of limiting who can connect to it, and what operations (reading or posting) they are allowed to do. Private servers typically allow access to anyone on the same network as the server, but public servers more often assign a user ID and password to each user. The user ID is typically just a string of letters and digits, unrelated to any e-mail address. The

server has no idea what e-mail address each user might use, and it makes no attempt to enforce any rule in the From: line of posted messages other than requiring a correct form of email address.

44. Servers typically add a header to messages that makes it possible to identify the user who posted a message, such as the X-Trace header in the message shown above, which is reproduced below.

```
X-Trace: gal.iecc.com 1309888575 66503 64.57.183.58 (5 Jul
2011 17:56:15 GMT)
```

Since my own server applied this header, I know that the fields are an internal version of the posting time, an internal process number I can check against my logs, and the IP address of the posting user. But that only applies to messages posted directly to this server.

45. Giganews adds an X-Trace: header to messages it originates that includes the posting user's user name, IP address, posting date, and Message-ID in an encrypted form. Since Usenet operators, such as myself, do not know what kind of encryption other Usenet operators use, and typically have no access to other operators' user databases, operators cannot use the X-Trace to identify or terminate a poster whose messages originate with another service.

46. Messages typically pass through many servers. For example, the message shown above passed through about a dozen servers on its way from my server news.iecc.com, where I posted it, to mx04.eternal-september.org, the public server where I retrieved a copy. Since most messages are received from other servers, the server's operator has no way to tell who might have posted a message that arrived from another server other than manually trying to contact the operator of the posting server and seeing if that operator is inclined to cooperate. Since there is no security on messages received from other servers, it is not unknown for malicious servers to include false or misleading Path: and X-Trace headers to disguise the actual server where a message originated.

47. News servers exchange messages directly via NNTP, as described above. The exchange is a fundamental part of Usenet; it allows a message posted at one server to be available to users at other Usenet servers around the world. Since NNTP is designed for high performance, it transfers messages unmodified from one server to the other. The sending server provides a list of Message-IDs of available messages, the receiving server replies to say which of those messages to send (typically declining only ones that it already has), and the sending server sends the desired messages. Once the relationship is established, a one-time manual process, the servers make no effort to validate messages from other servers. In particular, a server does not check the From: address or other author identification, but simply copies the message as offered.

48. For these reasons, Usenet services are not in a position to terminate postings by users of other services. Any repeat infringer termination policy can therefore apply only to a service's own subscribers or account holders. Usenet service providers may be able to identify their own users who post infringing messages, if they get Message-IDs to locate those messages. With respect to its own users, a Usenet service provider such as Giganews needs Message-IDs to locate messages and examine the X-Trace: header information to determine who sent them.

I declare that to the best of my knowledge the foregoing is true and correct.

Executed on November 25, 2013.

John Levine, Ph.D.